AD-A096 282 MINNESOTA UNIV MINNEAPOLIS DEPT OF AEROSPACE ENGINE--ETC F/G 12/1
STRUCTURAL INELASTICITY. XXVI. BEAMPLT: A PROGRAM FOR FINDING T--ETC(U)
JAN 81 P TAIT, P G HODGE N00014-75-C-0177
UNCLASSIFIED AEM-H1-26 NL

OF 1
AD
A096

END
DATE
FILMED
4-81
DTIC

Report AEM-H1-26

(12)

STRUCTURAL INELASTICITY XXVI

BEAMPLT: A Program for Finding the Yield-Point Load of Transversely Loaded Rectangular Grids

Patrick Tait, Undergraduate Research Assistant
Philip G. Hodge, Jr., Professor of Mechanics

Department of Aerospace Engineering and Mechanics
University of Minnesota
Minneapolis, Minnesota 55455

January 1981

Technical Report

Qualified requesters may obtain copies of this report from DDC.

Prepared for:

OFFICE OF NAVAL RESEARCH
Arlington, VA 22217

OFFICE OF NAVAL RESEARCH
Chicago Branch Office
536 South Clark St.
Chicago, IL 60605

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AEM-H1-26 (14)

2. GOVT ACCESSION NO.: AD-A096 282

3. RECIPIENT'S CATALOG NUMBER: 9

4. TITLE (and Subtitle): (6) STRUCTURAL INELASTICITY XXVI. BEAMPLT: A program for finding the yield-point load of transversely loaded rectangular grids.

5. TYPE OF REPORT & PERIOD COVERED: Technical Report.

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s): (10) Patrick Tait, Undergraduate Research Ass; Philip G. Hodge, Jr., Professor of Mech.

8. CONTRACT OR GRANT NUMBER(s): (15) N00014-75-C-0177

9. PERFORMING ORGANIZATION NAME AND ADDRESS: University of Minnesota, Minneapolis, Minnesota 55455

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: NR 064-429

11. CONTROLLING OFFICE NAME AND ADDRESS: OFFICE OF NAVAL RESEARCH, Arlington, VA 22217

12. REPORT DATE: (11) Jan[redacted]81

13. NUMBER OF PAGES: 45 (12) 25

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): OFFICE OF NAVAL RESEARCH, Chicago Branch Office, 536 South Clark St., Chicago, IL 60605

15. SECURITY CLASS. (of this report): Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

~~Qualified requesters may obtain copies of this report from DDC.~~

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Grids, Yield-point load, Collapse, Linear Programming.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

An interactive computer program for finding the yield-point load of a transversely loaded rectangular grid is described. The maximum statically-admissible multiplier and an associated moment distribution are found by Linear Programming.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

405 315

1 pg

BEAMPLT: A PROGRAM FOR FINDING THE YIELD-POINT LOAD OF TRANSVERSELY LOADED RECTANGULAR GRIDS

by

Patrick Tait

and

Philip G. Hodge, Jr.

1. Introduction. BEAMPLT is a FORTRAN computer program for finding the collapse load of transversely loaded structural grids. A grid consists of two sets of parallel beams at right angles to each other, as shown in Fig. 1; a given set of transverse loads $f_{i,j}$ (some of which may be zero) is applied to the nodes where two beams intersect. The objective is to find the maximum safety factor P of the grid, defined as that multiplier of the given loads such that the grid will just collapse under the loads $Pf_{i,j}$, but will support any smaller multiplier of the loads. An example of such a structure is found in many buildings where a floor is supported by a grid structure.

In the analysis of the problem, we will assume the grid is made of a perfectly-plastic material, and deformations are small, prior to collapse. We shall also neglect any torsional strength in the beams. The beams will have a maximum bending moment which can be transmitted across a beam section.

Under these assumptions the lower bound theorem of limit analysis states that P is the largest multiplier for which a statically admissible distribution of beam moments can be found. The moments must be everywhere in equilibrium with the loads $Pf_{i,j}$, and nowhere exceed the maximum bending moment (yield moment) of the beam.

BEAMPLT solves this problem by expressing it as a Linear Programming problem and calling on a previously prepared program LPKODE [1]* to solve it. BEAMPLT may be used directly or with a driver program PATIE, and it may be used interactively or in batch. Section 4 of this guide describes the program from the

*Numbers in square brackets refer to references listed on page 20.

user's point of view. First, however, in Sec. 2 we will describe the physical problem in more detail, and in Sec. 3 we will show it can be converted into one of Linear Programming. Section 6 will give the results of some typical problems, and Sec. 7 will look at the question of uniqueness.

2. <u>The Physical Problem</u>. As stated in the introduction, the problem which BEAMPLT aids in solving is a grid structure. A matrix notation, as shown in Fig. 2, will be used to discuss the problem and identify the moments.

We consider first a generic interior node i,j and derive a static equilibrium equation. Fig. 3 shows a free body diagram and defines the lengths and sign conventions. Each of the four beam segments at the node has shear forces and moments acting on it. Since torsional moments are neglected, the two horizontal moments $M_{i,j}$ at node i,j must be equal, as must the vertical moments $M'_{i,j}$. Out-of-plane and moment equilibrium of the beam segment ℓ_{j-1} require

$$V_1 = V_2 = (M_{i,j} - M_{i,j-1})/\ell_{j-1} \tag{1}$$

with similar expressions for each of the other segments.

Satisfying shear force equilibrium at the node, we have from Fig. 3,

$$V_2 + V_3 + V_6 + V_7 = Pf_{i,j} \tag{2}$$

where P is the load factor and $f_{i,j}$ is the load applied at the location i,j. Substituting the moment relations such as (1) in

the shear equation gives

$$\frac{M_{i,j} - M_{i,j-1}}{\ell_{j-1}} + \frac{M_{i,j} - M_{i,j+1}}{\ell_j} + \frac{M'_{i,j} - M'_{i-1,j}}{\ell'_{i-1}} +$$

$$\frac{M'_{i,j} - M'_{i+1,j}}{\ell'_i} = Pf_{i,j} \tag{3}$$

This equation holds for all nodes except near or on boundaries.

For nodes next to or on a boundary the procedure is the same, but the boundary condition defines some of the moments so that Eq. (3) will become simplified.

We will consider four possible boundary conditions along each side, plus the possibility of corner supports. The four possibilities are (a) free cantilever, (b) free cross beam, (c) simply supported, and (d) clamped. Figure 4 illustrates grids with various combinations of edge type.

Unless the beam is clamped, the moment at its end must be zero. Therefore, if the left side of the beam is case (a), (b), or (c); $M_{i,1} = 0$ for all i. Thus for j=2, one of the moments in Eq. (3) is replaced by zero. A similar remark applies to j=m-1, i=2, or i=n-1 when the right, top, or bottom edge, respectively, is anything other than clamped. However, if an edge is clamped, the moment is unknown and Eq. (3) for the neighboring column or row is unchanged.

For nodes on the boundary the changes are more significant. If the left edge is a free cantilever, then there are no vertical beams and, of course, there is no beam segment to the left of the node. Thus Fig. 3 reduces to Fig. 5a, and

Eq. (3) reduces to

$$-M_{i,2}/\ell_1 = Pf_{i,1} \tag{4a}$$

with similar simple equations if other edges are free cantilever.

On the other hand, if there is a free cross beam along the left edge, there will be three beam segments as shown in Fig. 5b, and Eq. (3) takes the form

$$\frac{-M_{i,2}}{\ell_1} + \frac{M'_{i,1} - M'_{i-1,1}}{\ell'_{i-1}} + \frac{M'_{i,1} - M'_{i+1,1}}{\ell'_i} = Pf_{i,1} \tag{4b}$$

Again, there are similar simplifications for the other edges.

If the end of a beam is clamped or simply supported, then vertical motion is prohibited and the support provides an external reaction force $R_{i,j}$. It follows that vertical equilibrium at such a node serves only to define $R_{i,j}$, and does not constitute a constraint on the moments. For our purposes, then, we do not enforce Eq. (3) on edges of type (c) or (d).

Since each edge may independently be of any of the four types, there would appear to be many different possibilities at the corners. However, if either edge at a corner is of type (c) or (d) (corners C, D, E, or H in Fig. 4) there will be no Eq. (3), and if both edges are type (a) (corner A) there are no beams at the corner, hence the load must be zero. If the top edge is a free cross beam and the left edge a free cantilever, (corner B) the upper left corner node is governed by Eq. (4a) with i=1. Other corners of this type have similar equations.

Finally, if both the top and left edge are cross beams, we

allow for two possibilities. If they are both free cross beams (corner G in Fig. 4), as shown in Fig. 5c, Eq. (4b) simplifies further to

$$-M_{1,2}/\ell_1 - M'_{2,1}/\ell'_1 = Pf_{1,1} \tag{4c}$$

with similar equations at other corners. The other possibility is that although the cross beams are otherwise free, the corner rests on a simple support as at F in Fig. 4. In this case, of course, Eq. (3) is not applicable.

Other modifications can be made in eq. (3) if the node is on a line of symmetry. Such lines may be vertical or horizontal, may be along a beam or between two beams, or may be a diagonal of the grid. We shall not detail the resulting forms of Eq. (3), but we note that vertical and/or horizontal lines of symmetry are incorporated in BEAMPLT and their use will be described in Sec. 4.

For any symmetric beam section the yield moment may be regarded as a known property of the material yield stress σ_0 and dimensions of the beam cross section. It corresponds to the top and bottom halves of the section being at the uniform stresses $-\sigma_0$ and $+\sigma_0$, respectively. For example, the yield moments for the rectangular and I sections (neglecting fillets) shown in Fig. 6 are, respectively,

$$M_0 = \sigma_0 wh^2/4 \tag{5a}$$

$$M_0 = \sigma_0 [t_w (h/2 - t_f)^2 - wt_f (h - t_f)] \tag{5b}$$

We assume that all horizontal beams have the same yield moment M_0 and all vertical yield moments are M'_0. Then the moments must satisfy

$$-M_0 \le M_{i,j} \le M_0 \tag{6a}$$

$$-M'_0 \le M'_{i-j} \le M'_0 \tag{6b}$$

Using the lower-bound theorem of limit analysis [2, Sec. 1-3] we may now state the problem of finding the collapse load as follows:

<u>Problem LB</u>

Find a distribution of moments $M_{i,j}$ and $M'_{i,j}$ which maximizes P subject to Eqs. (3), (4), and (6).

3. <u>Linear Programming</u>. A typical Linear Programming problem is concerned with a vector $(x_1, x_2, \ldots, x_N)$, a scalar objective function

$$\phi = \sum_{j=1}^{N} c_j x_j \tag{7a}$$

which depends linearly on x_j, and a set of constraints

$$\sum_{j=1}^{N} a_{ij} x_j (\cdot R_i \cdot) b_i \tag{7b}$$

where the symbol $(\cdot R \cdot)$ may stand for any of $=, \le$, or $\ge$. Equations (7b) must be written so that the b_i are all non-negative, and the vector components x_j are also subject to

$$x_j \ge 0 \tag{7c}$$

The Linear Programming problem may then be stated as

<u>Problem LP</u>

Find a vector x_j which maximizes ϕ subject to (7b), and (7c).

The problems LB and LP are similar but not identical. If the unknown moments $M_{i,j}$ and $M'_{i,j}$ are renamed as components of a vector x_k, then Eqs. (3), (4), and (6) will all have the form of (7b). However, it is not obvious how P can be made to correspond to ϕ, and the LB problem apparently has no requirement corresponding t (7c).

These difficulties are easily resolved. If each horizontal moment $M_{i,j}$ is related to a vector component x_k by

$$x_k = M_{i,j} + M_0 \tag{8a}$$

then (6a) becomes

$$0 \le x_k \le 2M_0 \tag{8b}$$

Clearly, the left-hand Inequality (8b) is the same as (7c) and right-hand one is a particularly simple form of (7b). Similarly for each unknown moment $M'_{i,j}$ we assign a vector component x_ℓ by

$$x_\ell = M'_{i,j} + M'_0 \tag{8c}$$

whence (6b) becomes

$$0 \le x_\ell \le 2M'_0 \tag{8d}$$

The objective function can be handled in either of two ways. On the one hand, a particular Eq. (3) for which $f_{i,j} \neq 0$ can be taken as the definition of P, (clearly in the same form as (7b)) and this expression for P substituted in the remaining Eqs. (3) to put them in the form of (7b). However, a simpler alternative is to introduce one additional new component x_z and define

$$P = x_z \tag{8e}$$

which is a particularly simple form of (7a). Replacement of P by x_z in all of (3) puts them in the form of (7b).

Thus, we have converted the LB problem into a LP problem. When the vector components are solved for, we can use (8) to convert them back to the load and moments. Appendix A shows the bookkeeping details of the conversion from a LB problem to a LP problem.

4. Using BEAMPLT. Clearly, to derive the equations for a fair sized problem and write the necessary information to LPKODE would take a considerable amount of work. Thus, the need for BEAMPLT.

BEAMPLT is a fairly simple program and easy to use. However, it does require some manipulation of files. Also, LPKODE's output is not easily related to the grid problem. Therefore, we have provided a procedure file entitled PATIE. PATIE will run BEAMPLT,

manipulate files, and interpret LPKODE's output. To use PATIE all the user needs to do is type in the control statement: CALL,PATIE.

PATIE is designed primarily for interative use and we shall describe it here from that viewpoint. Appendix C indicates the simple modification necessary to use it in batch.

As soon as PATIE has been called, it will begin to query the user for the necessary input for BEAMPLT. Basically, this information must define the size and spacing of the grid, the boundary conditions, the loads, the values of M_0 and M'_0, and the symmetries. PATIE will ask for this information with specific questions which are designed to be self-explanatory. Where appropriate it will branch on the answers.

We can most clearly discuss the input in relation to an example. To this end we consider the grid shown in Fig. 7a. The top edge is clamped, the side edges are simply supported, and the bottom edge is a free crossbeam. All beams are rectangular and made of carbon steel with a yield stress $\sigma_0 = 50 \cdot 10^3$ psi and a height of 1 inch. The horizontal beams are 1.2 inches wide and the vertical ones are 2.0 inches wide. Thus, it follows from Eq. (5a) that

$$M_0 = 15{,}000 \text{ lb-in} \qquad M'_0 = 25{,}000 \text{ lb-in} \tag{9}$$

The two interior nodes are each loaded with 200 pounds and no load is applied to the bottom nodes on the cross beam.

Section 5 shows the actual computer input and output for this example to which we have only added circled numbers to use

as reference. We will comment on specific points in relation to some of these numbers.

(2) The computer provides these dark squares so that the password remains private.

(4) Any consistent set of units may be used, but they must remain within these bounds.

(5) When two numbers are called for, they may be separated by a comma, one or more blank spaces, or both.

(6) If the response had been 0, the next question would have asked for the value of the uniform load.

(8) A 1 is entered only if (a) the grid spacing is symmetric, (b) the right and left boundaries are the same, and (c) the loads are symmetric. Each symmetry option reduces the problem to about half its previous size. In fact, the problem which is actually solved is the left half of the original as shown in Fig. 7b. The program automatically does the renumbering of nodes and makes the necessary modification in Eq. (3) for nodes 4 and 6 to allow for the symmetric right-hand boundary. However, the user must only enter loads which appear in this smaller half-grid. Note that with vertical symmetry we use the left half, with horizontal symmetry we would use the top half, and with both symmetries the small grid would be the top left quarter.

(10) This type of question will be asked for each degree of symmetry.

(11) This question will occur only if a 1 was entered in question (6); otherwise the computer will skip to question (13). Notice that there is only 1 load in the half-grid left after symmetry.

(12) If the answer to (11) were N, this question would be repeated N times.

(18) This question only refers to the original complete grid. Since vertical symmetry was assigned, the second and third numbers must be the same.

(19) If 5 had been entered the user would have been asked how many supports, and their location. Again, the user must only enter the supports which appear in the smaller half-grid.

(20) All input has now been given for BEAMPLT. BEAMPLT now carries out necessary computations and produces a file in the appropriate format for input to LPKODE. PATIE calls on LPKODE to solve the Linear Programming problem, reads the resulting solution, and converts it back into the form of results for the Lower Bound problem. The node numbering is that of Fig. 7b. The user, of course, can trivially expand the information back to the original grid. Figure 7c shows the results.

(21) The problem is now complete. If another problem is to be solved, the user again types CALL, PATIE and hits CR (Carriage Return). If this is the only or last problem the user types BYE and hits CR.

Instead of typing results the computer may respond with the message: YOUR ENTIRE FILE HAS BEEN DELETED. This message indicates the problem is too large to be solved interactively.

Naturally, errors are likely to be made when inputting data. Following the above example in Sec. 5 is another example

illustrating some typical types of input errors. Again, the example will be numbered for reference here.

(1) If an error is detected before the carriage on the typewriter is returned the user may back space one or more times and type over the error. For clarity, one may manually advance the paper one line before retyping.

(2) If the error is detected before the carriage on the typewriter is returned, the user may delete the line by hitting the ESC key, and then retype the correct input line.

(3) If a real value is entered where an integer is required the user is asked to retype the input correctly. Similarly with other typing errors such as hitting the letter "oh" instead of "zero".

(4) If only the first number is typed in when two or more are asked for, the computer will respond with another (?). Type the remaining number(s) only.

(5) If more numbers are typed than are asked for, the computer will simply ignore the extra numbers (2, 3, 4 in this example).

(6) An error detected after the carriage is returned (repetition of the previous load in this example) can not be corrected. Typing "stop" will abort the run and allow the user to start over by again typing CALL,PATIE.

5. Examples. We present here the actual computer input and output for the example discussed in the previous section. The program was run interactively.

Computer responses have been hand-labelled with circled

numbers for cross-reference to Sec. 4. After the computer has completed an instruction or question, it returns the carriage to a new line and types a question mark (?). The numbers following each (?) are typed in by the user in response to the question. The only other user-supplied part of the following output are the user code following the colon in (1), the user password typed over the dark squares in (2), and the words CALL,PATIE following the colon in (3). After each user response the "carriage-return" (CR) key should be hit to return control to the computer.

```
   80/09/02. 09.45.34. TERMINAL:  37, P 37
   UOFM CYBER 172 NOS 1.3 (08/31-BL).
1  USER NUMBER: FWN6090
   PASSWORD
2  [redacted]
   TERMINAL:   37, P 37/TTY
   USERS, TYPE *WRITEUP(NOTE)* 80/09/02.
3  RECOVER /SYSTEM: CALL,PATIE
4   NOTE:  THE MAGNITUDE OF THE YIELD MOMENTS MUST
    BE IN  THE RANGE:  1< MO < 50,000

    ALSO,THE APPLIED LOADS
    SHOULD BE LESS THAN 10,000

5   ENTER THE VERTICAL YIELD MOMENT,HORIZONTAL YIELD MOMENT
   ? 25000,15000
    IF THE LOAD IS UNIFORM ENTER THE INTEGER 0,
    IF NOT ENTER THE INTEGER 1
6  ? 1
7   ENTER THE NUMBER OF NODES ALONG THE TOP
   AND THE NUMBER OF NODES ALONG THE SIDE
   ? 4,3
8   IF THE GRID CONTAINS SYMMETRY ALONG A VERTICAL AXIS
    ENTER A 1,IF NOT ENTER A 0
   ? 1
9   IF THE GRID CONTAINS SYMMETRY ALONG A HORIZONTAL AXIS
    ENTER A 1,IF NOT ENTER A 0
   ? 0
10  IF THE NUMBER OF NODES ALONG THE TOP IS ODD
    ENTER A 1,IF NOT ENTER A 0
   ? 0
11  ENTER THE NUMBER OF NON-ZERO LOADS
   ? 1
12  ENTER THE ROW,THE COLUMN,AND THE LOAD APPLIED THERE
   ? 2,2,200
13  IF THE LENGTHS ALONG THE TOP ARE CONSTANT
    ENTER A 1, IF NOT ENTER 0
   ? 1
14  ENTER THE CONSTANT LENGTH
   ? 24
    IF THE LENGTHS ALONG THE LEFT SIDE ARE CONSTANT
15  ENTER A 1, IF NOT ENTER 0
   ? 0
16  ENTER THE BAR LENGTH BETWEEN ROWS   1 AND   3
   ? 36
17  ENTER THE BAR LENGTH BETWEEN ROWS   3 AND   5
   ? 12
```



```
    ASSIGNMENT NUMBERS FOR THE TYPE OF BOUNDARY CONDITIONS ARE
    1 - SIMPLY SUPPORTED
    2 - CLAMPED
    3 - FREE WITH CROSS BEAM
    4 - FREE CANTILEVER
18  ENTER THE ASSIGNMENT NUMBERS FOR THE TOP ,LEFT ,
    RIGHT,AND THE BOTTOM BOUNDARIES.
   ? 2,1,1,3
    IF THE CORNER OF AN INTERSECTION OF TWO TYPE 3 BOUNDARIES
19  HAS A SUPPORT ENTER 5, IF NOT ENTER 0
   ? 0
20  XEDIT 3.1.00
21  END OF FILE
    THE HORIZONTAL MOMENT AT NODE         4 IS        15000.0

    THE HORIZONTAL MOMENT AT NODE         6 IS        15000.0


    THE VERTICAL MOMENT AT NODE           2 IS       -25000.0

    THE VERTICAL MOMENT AT NODE           4 IS        7498.25

    THE LOAD FACTOR IS    10.7620
```

Below is an example of a few typical errors which may be made when inputting data. They are numbered for reference to the last part of Sec. 4.

```
60/12/29. 11.18.27. TERMINAL:   7, P  7
UOFA CYBER 1/2 NOS 1.3 (12/23-BR).
USER NUMBER: FGN6090
PASSWORD
[redacted]
TERMINAL:    7, P  7/IT:
RECOVER /SYSTEM: CALL,PATIE
 NOTE:  THE MAGNITUDE OF THE YIELD MOMENTS MUST
 BE IN  THE RANGE:  1< MO < 50,000

 ALSO,THE APPLIED LOADS
 SHOULD BE LESS THAN 10,000

1 ENTER THE VERTICAL YIELD MOMENT,HORIZONTAL YIELD MOMENT
? 1,2
    1
 IF THE LOAD IS UNIFORM ENTER THE INTEGER 0,
 IF NOT ENTER THE INTEGER 1
2 ? 0 *DEL*

 ENTER THE NUMBER OF NODES ALONG THE TOP
AND THE NUMBER OF NODES ALONG THE SIDE
3 ? 4.,4
 ERROR IN INPUT LINE - PLEASE RE-TYPE
? 4,4
 IF THE GRID CONTAINS SYMMETRY ALONG A VERTICAL AXIS
 ENTER A 1,IF NOT ENTER A 0
? 0
 IF THE GRID CONTAINS SYMMETRY ALONG A HORIZONTAL AXIS
 ENTER A 1,IF NOT ENTER A 0
? 0
 ENTER THE NUMBER OF NON-ZERO LOADS
? 2
4 ENTER THE ROW,THE COLUMN,AND THE LOAD APPLIED THERE
? 2
? 2,1
5 ENTER THE ROW,THE COLUMN,AND THE LOAD APPLIED THERE
? 2,2,1,2,3,4
 IF THE LENGTHS ALONG THE TOP ARE CONSTANT
6 ENTER A 1, IF NOT ENTER 0
? stop
 *TERMINATED*
/bye

FGN6090   LOG OFF    11.22.25.
A1JI007   SRU      0.564 UNTS.
```

6. <u>Results</u>. When the procedure file PATIE is used, the non-zero horizontal moments are printed first, followed by the vertical moments. The last value printed is the load factor.

The moments are identified by their node number. Examples of how to number a grid with or without symmetry were shown in Figs. 7a and 7b, respectively.

A number of problems have been solved. Four text book cases [2, Sec. 3-5] were run to test the program. They were a 3 by 3, a 4 by 4, a 5 by 5, and a 6 by 5 grid. They all were uniformly loaded with simply supported boundaries. Figure 8 shows the moments which have reached yield and the grid deformation profiles. To within round-off error the results are identical with those in Ref. [2].

A 5 by 5, a 6 by 5, and a 6 by 6 grid, all with a load applied at the second row and second column, and simply supported boundaries, were also run. Figure 9 shows results. Figure 10a shows a 6 by 6 grid with $M_0 \neq M'_0$, and non-uniform spacing. The same grid, but non-uniform loading is shown in Fig. 10b. Finally, Fig. 11 has the results of a 6 by 6 grid with clamped boundaries and a load applied at the second row and third column.

The figures identify those moments which have reached their yield point with a circle or square. A circle means the top of the beam is in compression and a square means the top is in tension. Notice that yield hinges can form only at yield moments, but that not all yield moments have hinges.

7. <u>Uniqueness</u>. A close look at some of the results in the previous section indicates that a solution may not be unique. The 5 by 5 grid of Fig. 8c is a good example. The grid,

boundaries, and loading are all symmetric, so a symmetric solution must exist. However, the computer has found a non-symmetric solution, indicating that the solution is not unique.

We also note that when a moment is numerically equal to M_0, we can permit a rotation of any magnitude in the direction of the moment. At the collapse load it must be possible to find enough such rotations to form a mechanism. In each of the figures we have shown such a kinematically admissible mechanism.

Again, looking at Fig. 8c we see that the yield moments shown in Fig. 8d are the only ones necessary for the deformation pattern. This solution is, of course, symmetric.

The question of uniqueness may also be viewed from our static equations. The grid in Fig. 8c has 19 unknowns, including the load factor, but only 9 equilibrium equations. Two or more inequalities may sometimes combine to force a unique value on a variable, but in general they only limit the range and cannot be counted on to determine the values uniquely. Therefore, we have twice as many unknowns as equations. Even if we take into account the mechanism, which in this case will define 6 of our unknowns, there are still more unknowns than equations. It is not surprising that the solution is non-unique.

This lack of uniqueness also explains the fact that in most of the examples there are several yield moments which are not yield hinges. The techniques used in solving the LP problem (see, for example [3]) include the selective replacement of some

inequalities as equalities so that we will generally end up with at least as many excess yield moments as there are degrees of indeterminacy.

In conclusion, we should note that if a grid were made of elastic-plastic beams, the solution would generally be unique. For each node where $|M| < M_0$ there would exist a linear relation between M and the rotation θ. The θ's can be expressed in terms of the nodal displacements w. Elimination of the w's would then give sufficient compatibility equations in M to determine a unique solution.

A major advantage of plastic methods is that we can directly find the information of most interest, such as the collapse load and those moments where yielding must occur, without finding all the details of an elastic-plastic solution.

References

1. T. Hoffman: LPKODE, University of Minnesota Computing Center
2. P. G. Hodge, Jr.: "Plastic Analysis of Structures," McGraw-Hill Book Co., New York, 1959.
3. S. I. Gass: "Linear Programming," Second Edition, McGraw-Hill Book Co., Inc., New York, 1964.

Appendix A

Conversion of LB to LP

To convert a LB problem to a LP problem, as done in Sec. 3, requires every unknown to be redefined in terms of a vector component x_j. The details of how this is done are presented here.

For an example we shall use the 4 by 4 grid, with clamped top boundary, free cross beam on the sides, and free cantilever on the bottom as shown in Fig. 12. We consider the horizontal moments first and number them by rows (top to bottom) and columns (left to right). We skip all moments which are known to be zero, hence in this example there are only four unknown moments, numbered 1 through 4 as shown in Fig. 12. We follow by numbering the unknown vertical moments (5 through 16 in this example), again going from left to right starting with the top row.

This type of numbering system can be done for all grids. Now, it is very easy to define a vector. If we abbreviate an unknown horizontal moment HM#, and an unknown vertical moment VM#, we then define a vector as follows;

$x_1 = HM1 + M_0$ $\qquad\qquad$ $x_5 = VM5 + M'_0$

$x_2 = HM2 + M_0$ $\qquad\qquad$ $x_6 = VM6 + M'_0$

.

.

.

$x_4 = HM4 + M_0$ $\qquad\qquad$ $x_{16} = VM16 + M'_0$

$x_{17} = P$ (P is the load factor)

This process can be used for any grid. Since our vector is defined we can write the inequalities as in Eq. (8c). Thus, we have converted a LB problem into a LP problem.

For an example of a typical equation we will look at the sixth node in Fig. 12, i.e., the node in column 2, row 2. The equilibrium equation before conversion is

$$\frac{M_{2,2} - M_{2,1}}{\ell_1} + \frac{M_{2,2} - M_{2,3}}{\ell_2} + \frac{M'_{2,2} - M'_{1,2}}{\ell'_1}$$

$$+ \frac{M'_{2,2} - M'_{3,2}}{\ell'_2} = Pf_{2,2} \tag{10a}$$

Noting that Eqs. *(8a) and (8c) can be written*

$$M_{i,j} = x_k - M_0 \tag{10b}$$

$$M'_{i,j} = x_\ell - M'_0 \tag{10c}$$

and $M_{2,1} = 0$ because of the left boundary condition, we can write (10a) as

$$(1/\ell_1 + 1/\ell_2)x_1 - x_2/\ell_2 + (1/\ell'_1 + 1/\ell'_2)x_{10} - x_6/\ell'_1$$

$$- x_{14}/\ell'_2 - f_{2,2}x_{17} = M_0/\ell_1 \tag{11}$$

which is in the form of (7b).

However, the yield moments will not always add together to make the right hand side positive. We recall that the right hand side must be non-negative. Therefore, if the right hand side should happen to be negative we must take the negative of both *sides of the equation. For example*, using exactly the same technique on node 8 in Fig. 12, we would obtain

$$-x_2/\ell_3 - x_8/\ell_1' + (1/\ell_1' + 1/\ell_2')x_{12} - x_{16}/\ell_2'$$

$$- f_{2,4}x_{17} = -M_0/\ell_3 \tag{12}$$

Since the right hand side is negative, we cannot use Eq. (12) as it stands but must replace it by

$$x_2/\ell_3 + \ldots\ldots$$

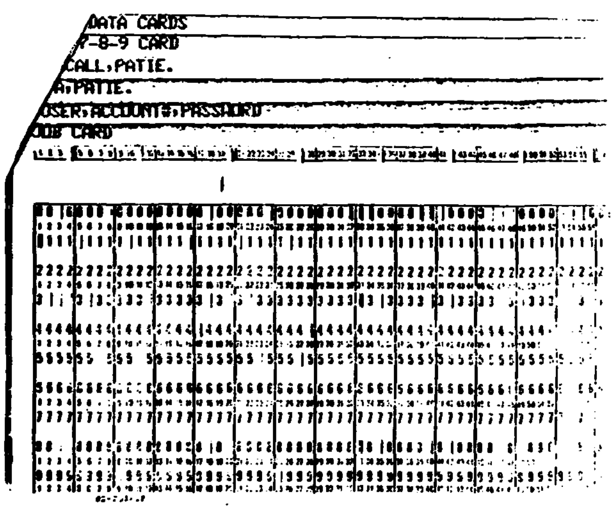
$$\ldots\ldots = M_0/\ell_3 \tag{13}$$

Appendix B

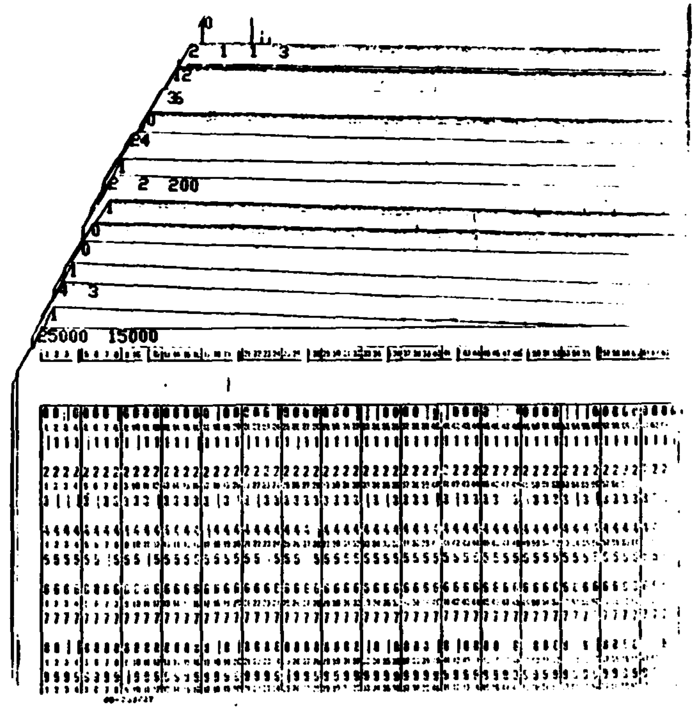
Using Batch Cards

As stated earlier LPKODE has a limit in the size of problem it can solve. This limit is even smaller when using timesharing. Thus, there may arise a need to use batch cards.

To run PATIE using cards the user must have a set of computer cards. The first card is a job card followed by a card with the account number and password. The next two cards should acquire PATIE and run the program. A 7-8-9 card is next. The data cards are last. The deck should look like this:



Each data card must contain the answer(s) to one and only one question as asked in timesharing, and they must be in the same order. As in timesharing, multiple data on a card may be separated by a comma, one or more spaces, or both. Since all data cards must be submitted at once, the user must anticipate all branches such as symmetry, number of loads, etc., and must continue with the data appropriate to the chosen branch. We use the problem in Sec. 4 to illustrate how the data deck should look:



Appendix C

Input Variables List

vertmo - the vertical yield moment

horzmo - the horizontal yield moment

loaddet - determines whether or not the load is uniform

fload - the value of the uniform load

n - the number of nodes along the top

m - the number of nodes along a side

isymv - determines if a vertical line of symmetry exists

isymh - determines if a horizontal line of symmetry exists

ioddv - determines whether the number of nodes along the top is odd or even

ioddh - determines whether the number of nodes along the side is odd or even

num - number of non-zero loads

i - the row number

j - the column number

force - the value of a non-uniform load

ltop - determines whether or not the lengths along the top are constant

clent - the value of the constant bar length along the top

horzl(j) - the value of a bar length in column "j"

lleft - determines whether or not the bar lengths along a side are constant

clenl - the value of the contant bar length along a side

vertl(i) - the value of the bar length in row "i"

itop - the assignment number for the top boundary

left - the assignment number for the left boundary

irt - the assignment number for the right boundary

ibtm - the assignment number for the bottom boundary

iisupt - determines whether or not a corner support is used

icorner - the number of times a corner support is used

nod - the node number where a corner support exists

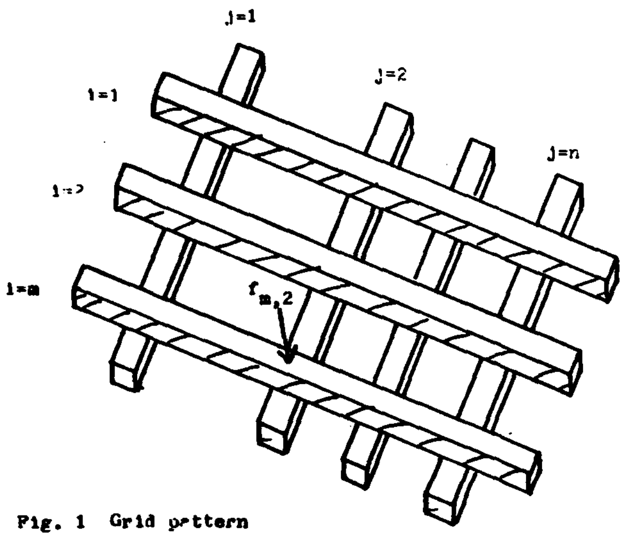


Fig. 1 Grid pattern

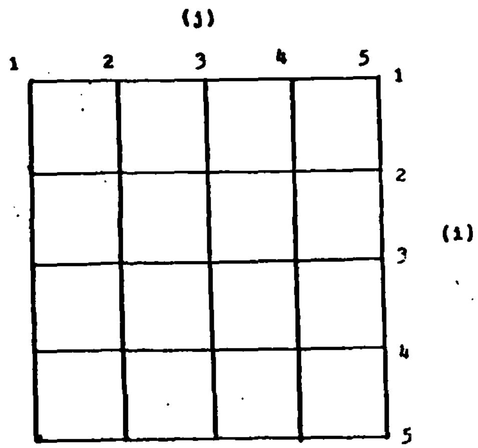


Fig. 2 Matrix notation system

A grid can have "m" rows and "n" columns. "i" represents the i^{th} row and "j" represents the j^{th} column.

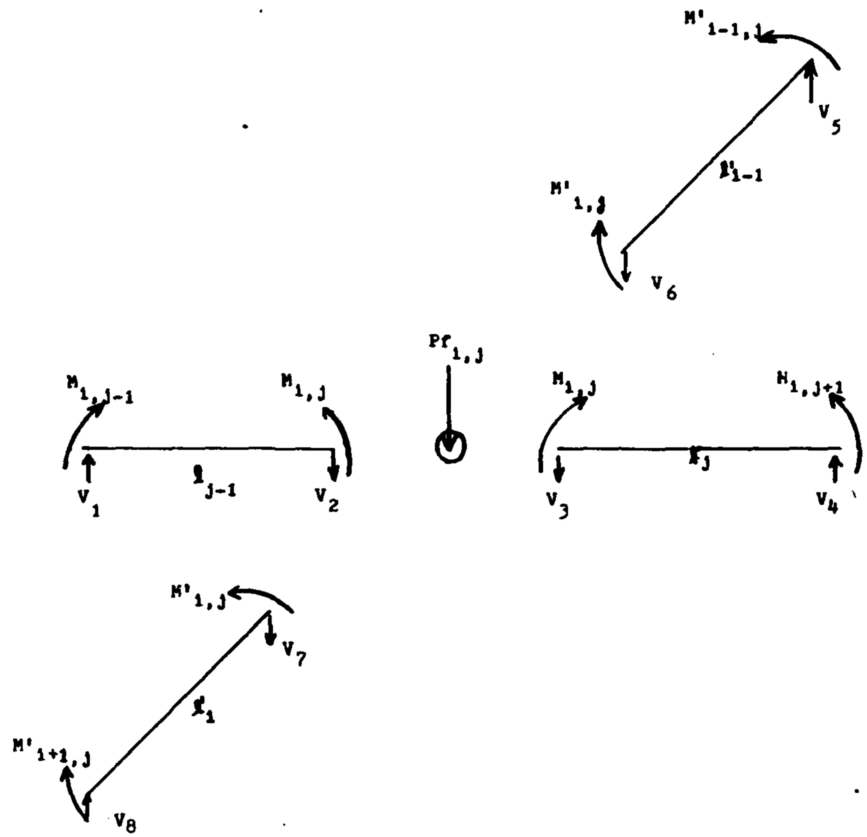


Fig. 3 Free body diagram of an interior node

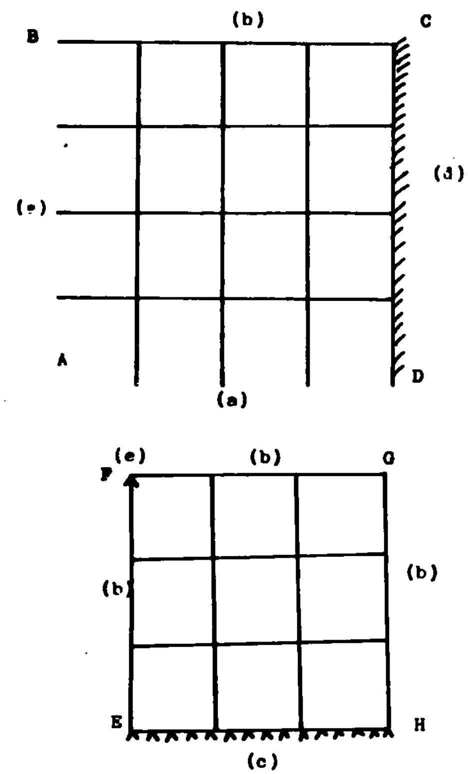
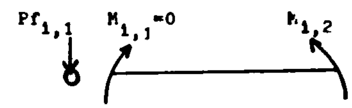
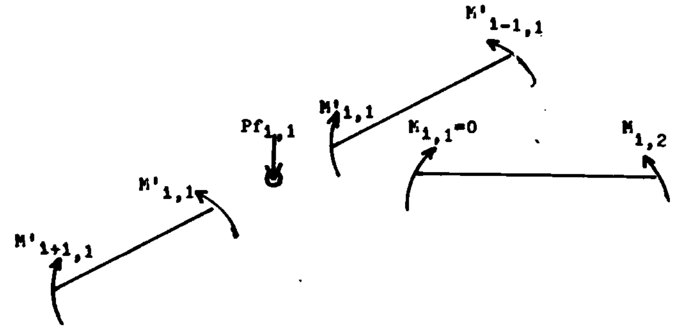


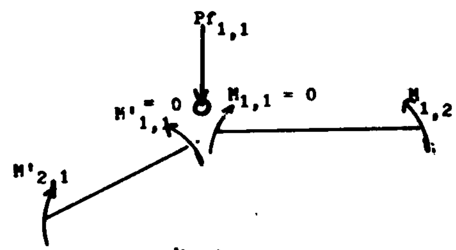
Fig. 4 Boundary conditions
(a) free cantilever
(b) free crossbeam
(c) simply supported
(d) clamped
(e) supported corner



(a) free cantilever

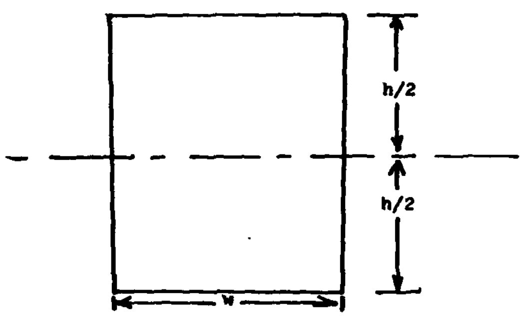


(b) free cross beam

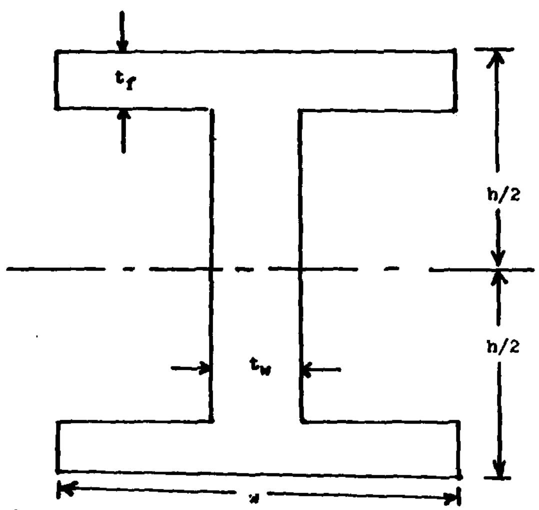


corner with two
(c) corner with two free cross beams

Fig. 5 Boundary relations

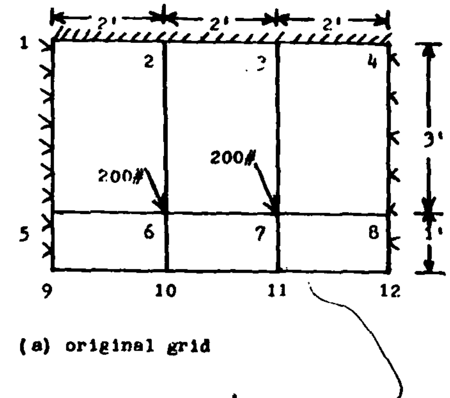


(a) rectangle

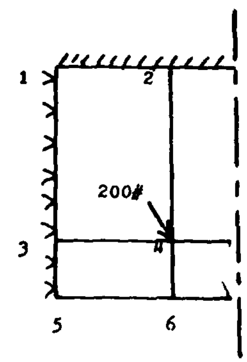


(b) I - beam

Fig. 6 X - sections

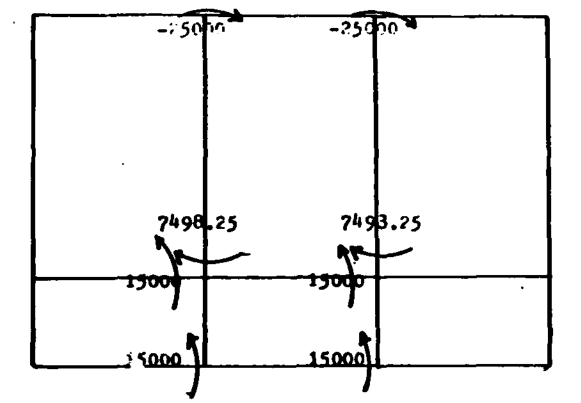


(a) original grid

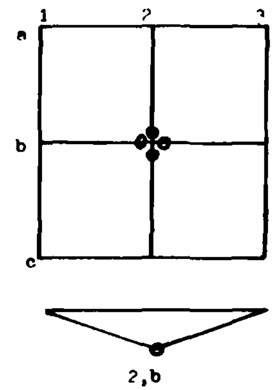


(b) smaller grid

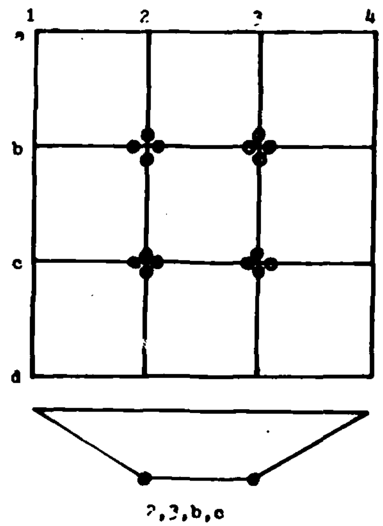
Fig. 7 Example Problem



(c) results   The load factor P is 10.762



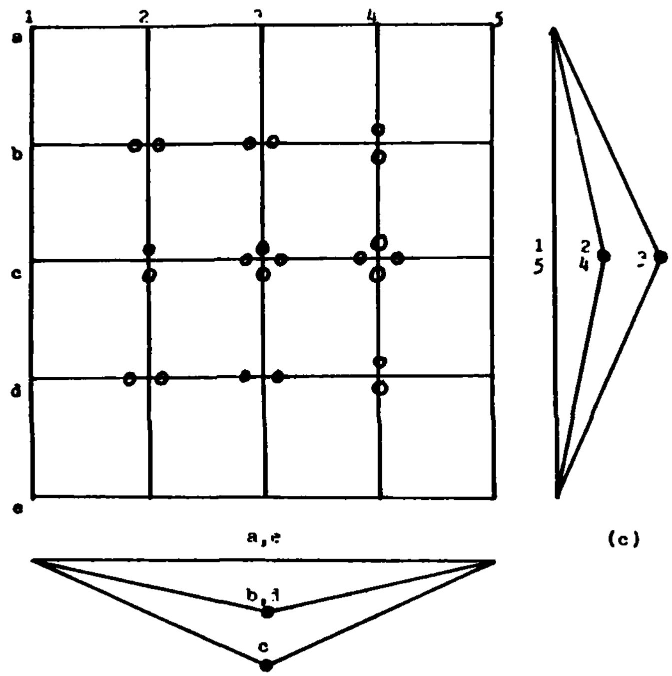
(a)



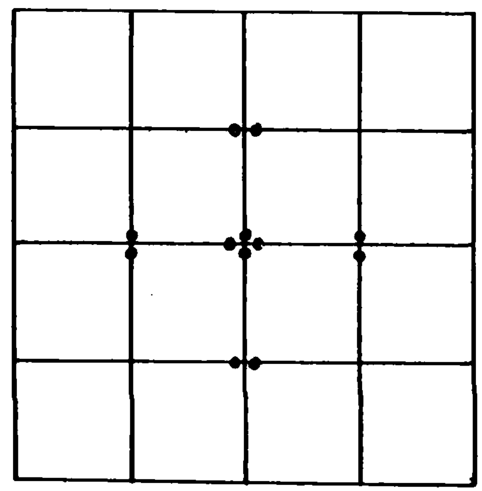
(b)

Fig. 8 Simply supported grids with a unit load applied to every interior node[2].

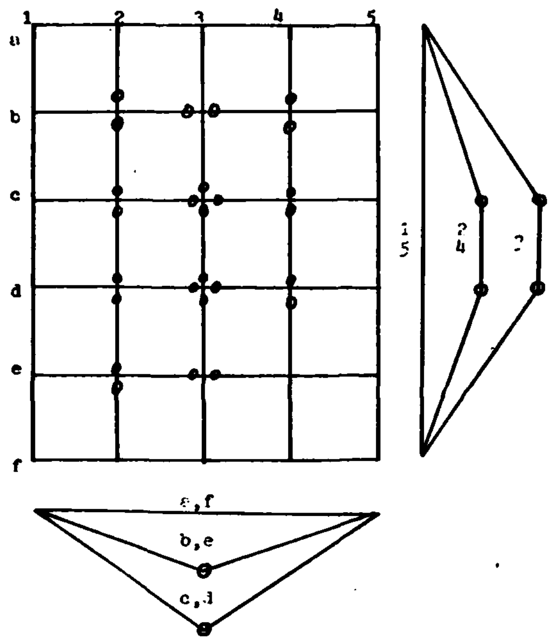
(a) 3 by 3 (P = 4)
(b) 4 by 4 (P = 4)
(c) 5 by 5 (P = 0.999 - 1)
(d) 5 by 5 using symmetry
(e) 6 by 5 (P = .8??)



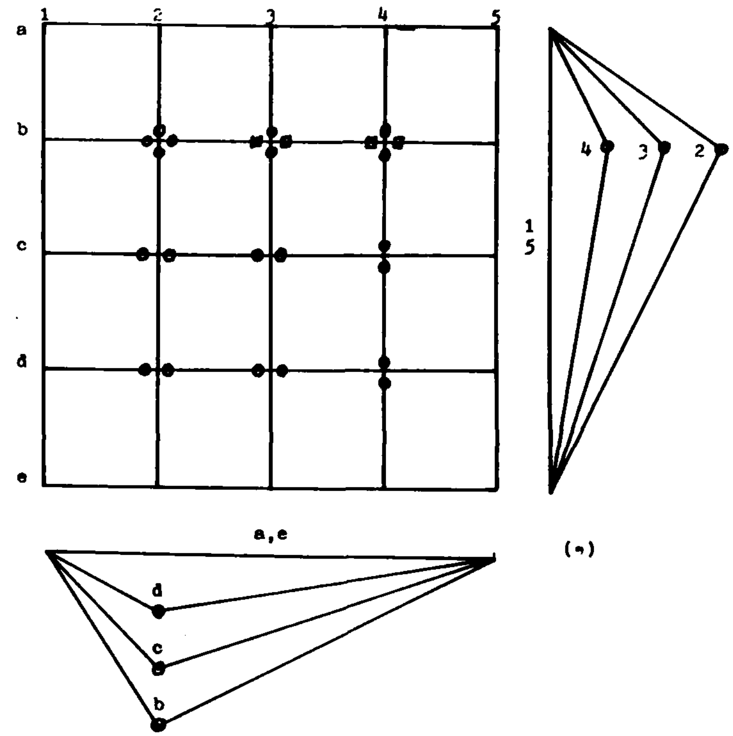
(c)



(d)



(a)



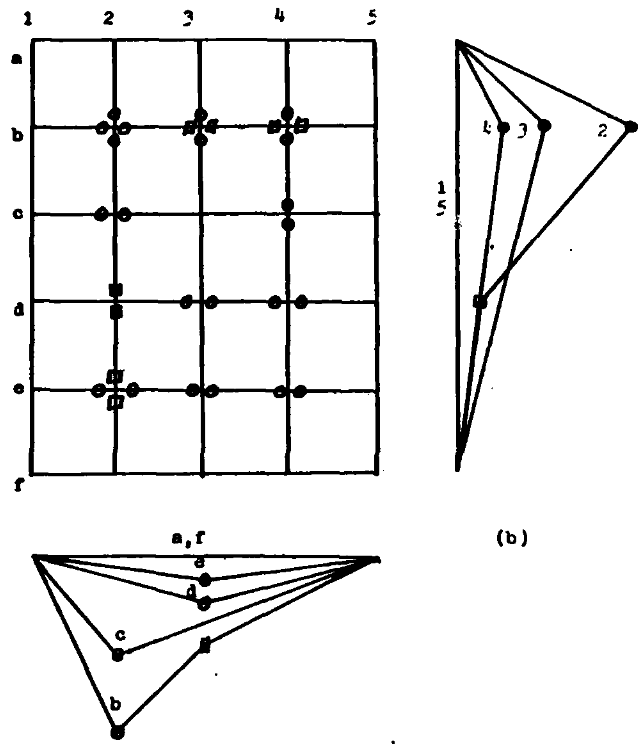
(a)

Fig. 9. Simply supported grids with unit load applied only to node b2. Small circles and squares represent moments equal to positive and negative yield moments, respectively.

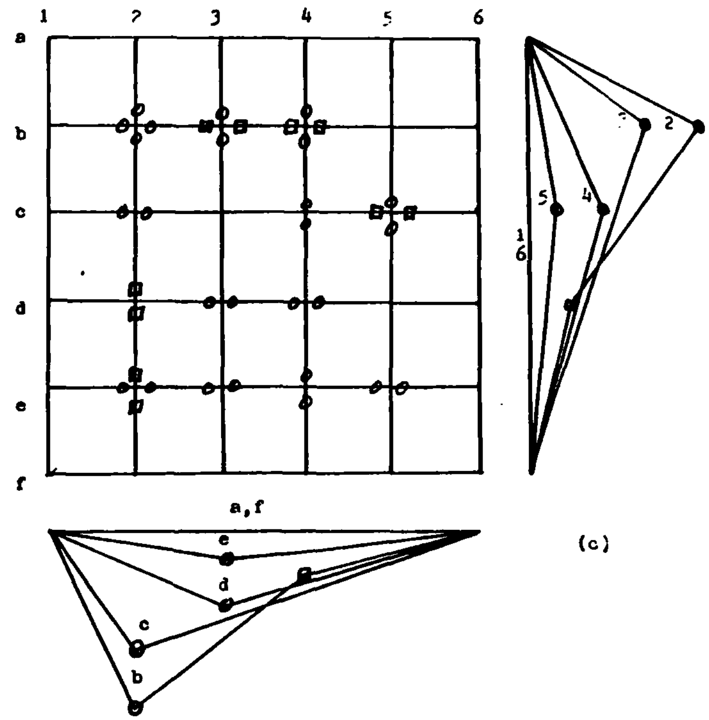
(a) 5 by 5 (P = 5.333)

(b) 6 by 5 (P = 5.6?3)

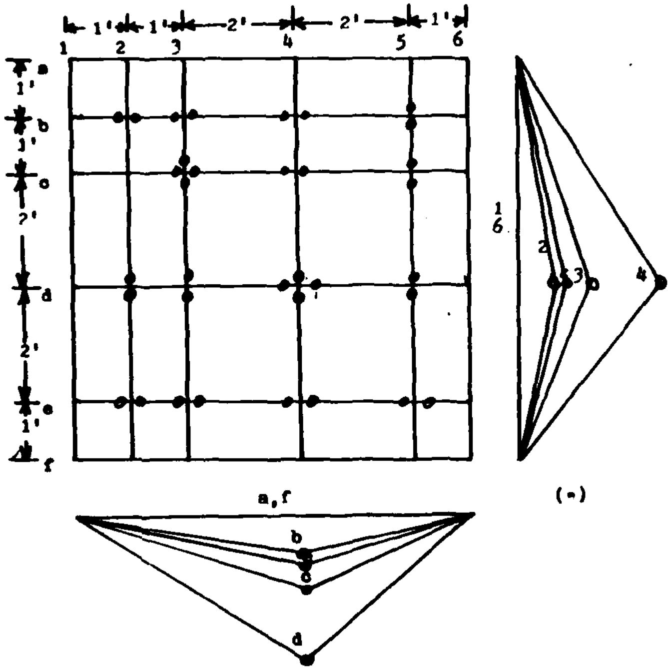
(c) 6 by 6 (P = 5.714).



(b)



(c)



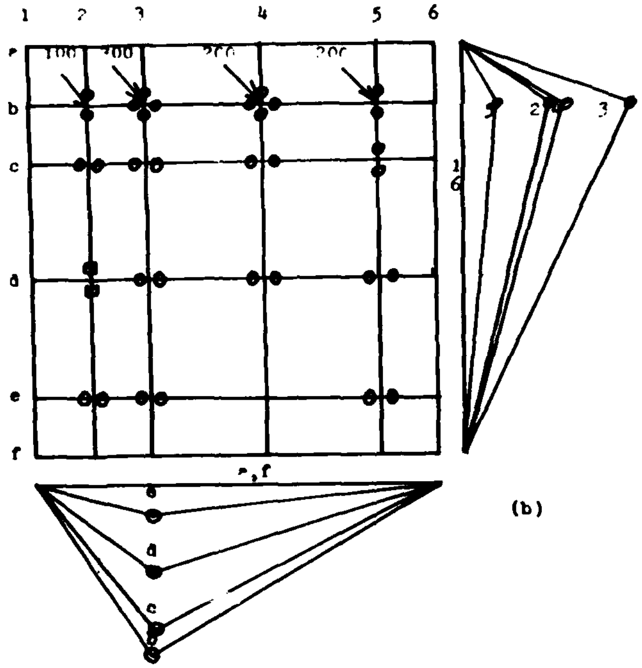


Fig. 10 6 by 6 simply-supported grid with M_0=15,000 lb-in,
M'_0=25,000 lb-in
(a) load of 100 lb at each interior node (P = 9.335)
(b) loaded as shown (P = 15.75)

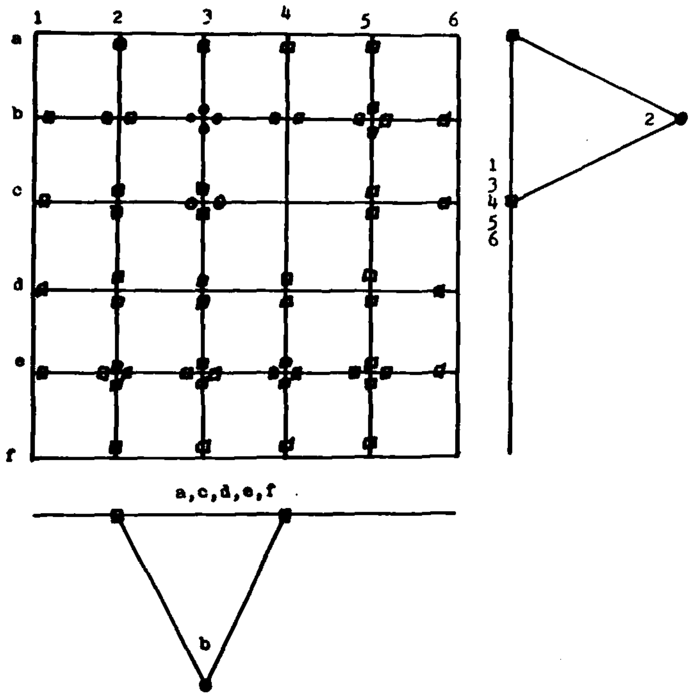


Fig. 11 Clamped 6 by 6 grid with unit load at b?
(P = 7.999 ≈ 8

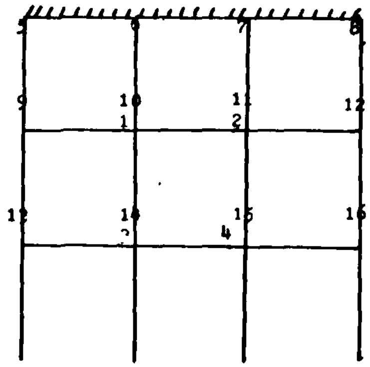


Fig. 12 Vector conversion

LMED

8